UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/695,195	10/25/2000	Satoru Fujita	029471/0145	1641

22428 7590 05/06/2004

FOLEY AND LARDNER
SUITE 500
3000 K STREET NW
WASHINGTON, DC 20007

EXAMINER	
JACOBS, LASHONDA T	

ART UNIT	PAPER NUMBER
2157	8

DATE MAILED: 05/06/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No. 09/695,195	Applicant(s) FUJITA ET AL.	
	Examiner LaShonda T. Jacobs	Art Unit 2157	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *16 March 2004*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-31* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-31* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Response to Amendment

This Office Action is in response to Applicants' amendment filed on March 16, 2004. Claims 1-31 are presented for further examination.

Claim Rejections - 35 USC § 103

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2. Claims **1-31** are rejected under 35 U.S.C. 103(a) as being unpatentable over Putzolu et al (hereinafter, "Putzolu', 6,681,243) in view of Takewaki et al (hereinafter, "Takewaki", 6,539,416).

As per claims **1**, **11**, **21** and **22**, Putzolu discloses a system in which a plurality of computers are interconnected over a network, a distributed application control system, wherein an agent platform owned by each computer comprises:

(a) an agent base (col. 10, lines 56-65), said agent base having:

(a1) a shell agent responsive to an input of a script language controlling the distributed application to interpret and execute said script language (col. 11, lines 4-26, line 67, col. 12, lines 1-14 and lines 49-66);

(a2) a local service agent furnishing information of a local file system to said computer (col. 15, lines 36-54); and

(a3) an application agent controlling the application (col. 6, lines 63-67, col. 7, lines 30-60 and col. 9, lines 48-56);

(a4) said agent base furnishing respective fields of execution to said shell agent, local service agent and the application agent (col. 10, lines 56-65 and col. 11, lines 1-30);

(c) a remote call module furnishing functions for an agent to have communication with an agent of an own computer or at least one other computer (col. 10, lines 35-53); and

(e) wherein execution of the application distributed over each computer is controlled responsive to an input of said script language (col. 10, lines 56-65 and col. 11, lines 1-30).

However, Putzolu does not explicitly disclose:

(b) agent mover furnishing the function of causing an agent to move to an agent to move base of at least one other computer; and

(d) agent generator generating an application agent.

In an analogous art, Takewaki discloses a managing system of a mobile agent system, the mobile agent system including:

(b) agent mover furnishing the function of causing an agent to move to an agent base of at least one other computer (abstract, col. 1, lines 47-67, col. 2, lines 1-16, col. 7, lines 35-59, col. 8, lines 59-67 and col. 9, lines 1-11); and

(d) agent generator generating an application agent (col. 6, lines 3-25).

Given the teaching of Takewaki, it would have been obvious to one of ordinary skill in the art to modify Putzolu by including an agent mover and an agent generator in order to allow agents to move from one computer to another in a timely and efficient manner.

As per claims **2** and **12**, Putzolu discloses wherein said shell agent includes:

- a current directory supervising a current directory in executing the script language (col. 15, lines 36-54); and
- a shell interpreter having syntax analyzer analyzing the syntax of said script language and command analysis- executor analyzing and executing a command (col. 11, lines 4-26, line 67, col. 12, lines 1-14 and lines 49-66); and
- wherein said current directory indicates an agent base on a remote controller (col. 15, lines 36-54), said shell agent itself is moved by said agent mover to agent controller to execute an application agent (col. 11, lines 4-26, line 67, col. 12, lines 1-14 and lines 49-66).

As per claims **3** and **13**, Putzolu discloses:

- wherein said shell agent further includes a repository path table of an agent (col. 7, lines 61-67 and col. 8, lines 1-12); and
- wherein the application agent is retrieved from the repository path and executed (col. 7, lines 61-67, col. 8, lines 1-12 and col. 12, lines 49-66).

As per claims **4** and **14**, Putzolu discloses:

- wherein said shell agent further includes a status table for supervising the status of an agent (col. 7, lines 61-67, col. 8, lines 1-12 and col. 12, lines 49-66);
- said status table has a parallel execution counter for an agent and terminal execution flag (col. 7, lines 61-67, col. 8, lines 1-12 and col. 12, lines 49-66); and
- wherein if, when the current base of said current directory is a remote computer, the status of the agent is not the parallel execution nor the terminal execution mode, said

shell agent is moved to said remote computer (col. 7, lines 61-67, col. 8, lines 1-12 and col. 12, lines 49-66).

As per claims **5** and **15**, Putzolu discloses:

- means for invoking a local service agent assisting each computer to access local resources for generating an application agent by said shell agent (col. 6, lines 63-67, col. 7, lines 30-60 and col. 9, lines 48-56);
- wherein when said application agent is generated, said shell agent generates the application agent through said local service agent instead of directly invoking said agent generator (col. 6, lines 63-67, col. 7, lines 30-60 and col. 9, lines 48-56).

As per claims **6** and **16**, Putzolu discloses:

- wherein said shell agent further includes an agent referencing table (col. 7, lines 61-67, col. 8, lines 1-12 and col. 12, lines 49-66); and
- wherein a request for movement or command execution is made to an application agent stored in said agent referencing table and booted outside (col. 7, lines 61-67, col. 8, lines 1-12, col. 11, lines 49-53 and col. 12, lines 49-66).

As per claims **7** and **17**, Putzolu discloses:

- wherein said application agent includes application booting means dependent on said computer(col. 7, lines 61-67, col. 8, lines 1-12, col. 11, lines 49-53 and col. 12, lines 49-66); and
- wherein said application is generated and supervised under commands by said shell agent (col. 7, lines 61-67, col. 8, lines 1-12, col. 11, lines 49-53 and col. 12, lines 49-66).

As per claims **8** and **18**, Putzolu discloses:

- wherein said application agent includes means for referencing an extension-application associating table to select an application to be booted depending upon the sort of the file extension (col. 6, lines 63-67, col. 7, lines 30-60 and col. 9, lines 48-56);
- said selected application being generated and supervised under a command of said shell agent (col. 6, lines 63-67, col. 7, lines 30-60 and col. 9, lines 48-56).

As per claims **9** and **19**, Putzolu discloses:

- wherein said agent platform further includes thread generator (col. 10, lines 56-65 and col. 11, lines 1-30); and
- wherein if parallel execution is specified in the input script language, as many threads as are required for executing said script language are generated to control the parallel execution of the application (col. 10, lines 56-65 and col. 11, lines 1-30).

As per claims **10** and **20**, Putzolu discloses:

- wherein said agent platform includes a sub-shell agent in an agent base of said remote controller when generating an application agent in said remote computer by the generated thread (col. 7, lines 61-67, col. 8, lines 1-12 and col. 12, lines 49-66);
- said sub-shell agent taking over the role of generating said application agent (col. 7, lines 61-67, col. 8, lines 1-12 and col. 12, lines 49-66).

As per claim **23**, Putzolu discloses a computer connected to one or more other computers over a network, said computer comprising:

- an agent platform, a file system local to said computer, an application and an agent repository; said agent platform including (col. 10, lines 56-65):

(a) an input unit receiving an input to said computer, an output unit issuing an output from said computer (col. 8, lines 13-30),

(b) a shell agent interpreting and executing an input script language to generate an application agent and to have communication (col. 7, lines 61-67, col. 8, lines 1-12 and col. 12, lines 49-66),

(c) a local service agent, as an agent for providing said computer with inherent information or functions, said local service agent supervising a local file system to provide the shell agent with information (col. 15, lines 36-54),

(d) an application agent executing a given task on request from said shell agent (col. 6, lines 63-67, col. 7, lines 30-60 and col. 9, lines 48-56);

(e) an agent base furnishing an area for storage of an agent program being executed on each agent (col. 10, lines 56-65);

(g) a thread generating unit furnishing a function of generating a new thread when an agent in an agent base is operated in a multi-thread operation (col. 10, lines 56-65 and col. 11, lines 1-30);

(h) a remote call module providing an agent in said agent base with a function of invoking a method from another agent in said agent base or an agent of another agent platform (col. 10, lines 35-53); and

- wherein said script language input through said input unit is interpreted by said shell agent to boot said application agent, said application agent supervising an actual application (col. 7, lines 61-67, col. 8, lines 1-12 and col. 12, lines 49-66);

- said shell agent and the application agent being movable to at least one other computer through said network, using said agent movement module, said shell gent being able to communicate with an agent in at least one other computer through said network (col. 7, lines 61-67, col. 8, lines 1-12 and col. 12, lines 49-66).

However, Putzolu does not explicitly disclose:

(f) an agent movement module providing an agent in said agent base with a function of movement to another agent platform; and

(i) an agent generating module invoked in generating a new agent in said agent base and executing the generated agent, said agent generating module referencing said agent repository to retrieve an agent program to generate a new agent in said agent base based on the retrieved result.

In an analogous art, Takewaki discloses a managing system of a mobile agent system, the mobile agent system including:

(f) an agent movement: module providing an agent in said agent base with a function of movement to another agent platform (abstract, col. 1, lines 47-67, col. 2, lines 1-16, col. 7, lines 35-59, col. 8, lines 59-67 and col. 9, lines 1-11); and

(i) an agent generating module invoked in generating a new agent in said agent base and executing the generated agent, said agent generating module referencing said agent repository to retrieve an agent program to generate a new agent in said agent base based on the retrieved result (col. 6, lines 3-25)

Given the teaching of Takewaki, it would have been obvious to one of ordinary skill in the art to modify Putzolu by including an agent mover and an agent generator in order to allow agents to move from one computer to another in a timely and efficient manner.

As per claim **24**, Putzolu discloses:

- wherein a shell interpreter in said shell agent is fed from said input unit with a script language and interprets the input script language, said shell interpreter executing the parallel processing if the script language is a parallel execution sentence (col. 7, lines 61-67, col. 8, lines 1-12 and col. 12, lines 49-66); and
- wherein if the script language can be processed as an internal command, said shell interpreter performs control to effect parallel processing, whereas, if otherwise, said shell interpreter performs control to effect external command processing (col. 7, lines 61-67, col. 8, lines 1-12 and col. 12, lines 49-66).

As per claim **25**, Putzolu discloses:

- said shell agent includes a shell interpreter having a syntax analysis unit responsive to an input of a script language from said input unit to interpret the input script language and a command analysis-executing unit for executing a command, a status table for supervising the state of execution of said shell agent, and a current directory storage unit for memorizing the current directory of said shell agent (col. 7, lines 61-67, col. 8, lines 1-12 and col. 12, lines 49-66); wherein
- said syntax analysis unit referencing a keyword table having a keyword registered thereon said syntax analysis unit when receiving an input having a syntax beginning with a keyword of a parallel executing sentence requesting said thread generating unit

to generate a new thread to effect parallel execution (col. 7, lines 61-67, col. 8, lines 1-12 and col. 12, lines 49-66);

- said command analysis- executing unit verifies whether or not an input command is an internal command that can be processed in said shell agent; if, as a result of verification, the input command is the internal command, a command definition is loaded from an internal command definition storage unit, memorizing the internal command definition, to execute the command definition; said agent movement unit being invoked if, in executing a directory movement command from among internal commands loaded from the internal command definition storage unit, movement of said shell agent itself is necessary (col. 7, lines 61-67, col. 8, lines 1-12 and col. 12, lines 49-66);
- said remote call module being utilized if a method of another agent is executed;
- the result of command execution by said command analysis executing unit being delivered to said output unit and output from said computer (col. 6, lines 1-28 and lines 48-67);
- an agent program for executing an external command being retrieved from an agent repository in an external command execution unit provided in said shell agent in case of execution of an external command; a repository path table being referenced to determine which path in said agent repository is to be retrieved or not to sequentially retrieve the agent repository for paths stored in said repository path table (col. 5, lines 1-6, lines 15-25, col. 55-67, col.. 6, lines 1-28, lines 48-67 and col. 7, lines 1-42);

- wherein if, as a result of retrieval of said agent repository, the necessary agent is retrieved, said external command execution unit requests the agent generating unit to generate an agent, said agent generating unit so-requested then generating an application agent (col. 5, lines 1-6, lines 15-25, col. 55-67, col.. 6, lines 1-28, lines 48-67 and col. 7, lines 1-42); and
- wherein when making a method call to an agent other than said shell agent, said external command execution unit invokes said remote call module (col. 5, lines 1-6, lines 15-25, col. 55-67, col.. 6, lines 1-28, lines 48-67 and col. 7, lines 1-42).

As per claim **26**, Putzolu discloses:

- wherein said local service agent receives an external command name and the path information stored in said repository path table from said external command execution unit to retrieve said agent repository and to invoke an actual agent generating unit to generate said application agent (col. 4, lines 40-67, col. 6,lines 1-28, lines 48-67 and col. 7, lines 1-3).

As per claim **27**, Putzolu discloses:

- wherein, if said internal command is a command indicating directory movement, the directory of destination of movement is extracted from said internal command to develop the extracted directory of destination of movement in a directory constituent element to store the base name and the directory in a base in a current base of the directory storage table and in an in-base directory, respectively (col. 7, lines 61-67, col. 8, lines 1-12 and col. 12, lines 49-66);

- the value of a parallel execution counter for storage of depth information of parallel execution, provided in said status table, is checked as to whether or not it is equal to 0; if the parallel execution counter is other than 0, it indicates that the shell agent is executing in parallel; the value of said current directory storage unit is updated to the value of the directory of the destination of movement specified by a command representing said directory movement, with the shell agent not being moved between computers (col. 7, lines 61-67, col. 8, lines 1-12 and col. 12, lines 49-66);
- if the parallel execution counter is 0, the shell agent is executing in a sole thread, so a terminal mode flag, provided in said status table, and which i s set to a first value and to a second value if the shell agent is being executed as a she l I awaiting an input from a terminal and if the shell agent is being executed as a batch command without being connected to a specified terminal, respectively, is referenced, with movement between computers of the shell agent by a command representing the directory command not being made, but only the updating of the value of the current directory storage unit being made, when said terminal mode f lag is of the first value (col. 7, lines 61-67, col. 8, lines 1-12 and col. 12, lines 49-66); and
- wherein if said terminal mode flag is of a second value, the batch mode is being executed, so that the value of the current directory storage unit and the directory of destination of movement are compared to each other; if the current directory storage unit and the directory of destination of movement are on the same computer, value of the current directory storage unit is updated; if the current directory storage unit and the directory of the destination of movement are on the same computer, said shell agent is

moved in a controlled manner to an agent base of a computer where there exists a current base of the directory of destination of movement (col. 7, lines 61-67, col. 8, lines 1-12 and col. 12, lines 49-66).

As per claim **28**, Putzolu discloses:

- wherein in detecting a parallel execution sentence to execute the parallel execution processing in a syntax analysis unit of said shell interpreter (col. 7, lines 61-67, col. 8, lines 1-12 and col. 12, lines 49-66),
- a parallel execution counter of said status table is incremented by one to record that the shell agent is currently in parallel executing state (col. 7, lines 61-67, col. 8, lines 1-12 and col. 12, lines 49-66);
- as many threads as there are commands in said parallel executing sentence are generated by said thread generating unit (col. 10, lines 56-65 and col. 11, lines 1-30);
- the generated number of threads is recorded and set in a thread variable (col. 10, lines 56-65 and col. 11, lines 1-30);
- the commands in said parallel executing sentence are executed in parallel in respective threads (col. 7, lines 61-67, col. 8, lines 1-12 and col. 12, lines 49-66);
- on completion of the execution of the respective threads, the number of values of thread variables is decremented by one, with the number of said threads being zero at a time point of end of all threads to terminate parallel execution (col. 10, lines 56-65 and col. 11, lines 1-30); and

- wherein the value of said parallel execution counter is decremented by one to terminate the execution of the entire parallel execution (col. 7, lines 61-67, col. 8, lines 1-12 and col. 12, lines 49-66).

As per claim **29**, Putzolu discloses:

- wherein said external command execution unit checks whether or not the current base stored in said current directory storage unit actually is coincident with the current base actually executing the shell agent; in case of coincidence, said external command execution unit checks whether or not a program of an application agent in question is present in an agent repository path of the agent repository being executed in order to generate an external application agent in the base being executed; if there is such program of the application agent, the external command execution unit asks said agent generating unit to load the application agent program from the agent repository to execute the application agent (col. 7, lines 61-67, col. 8, lines 1-12 and col. 12, lines 49-66);
- in case of non-coincidence of the current base stored in said current directory storage unit with the base currently executing the shell agent, said remote call module is used to generate a sub-shell agent in said current base (col. 7, lines 61-67, col. 8, lines 1-12 and col. 12, lines 49-66); and
- wherein it is checked whether or not there is a program of an application agent in question in an agent repository path of the agent repository of the current base in said sub-shell agent; in case such program of said application agent exists, the program of the application agent from said agent repository is loaded to ask the agent generating unit to load the

application agent program from said agent repository to execute the application agent (col. 7, lines 61-67, col. 8, lines 1-12 and col. 12, lines 49-66).

As per claim **30**, Putzolu disclose:

- wherein it is checked whether or not the current base stored in said current directory storage unit coincides with the executing base actually executing the shell agent; in case of coincidence, said local service agent of the executing base is asked to execute the external agent to generate the external application agent in the executing base (col. 7, lines 61-67, col. 8, lines 1-12 and col. 12, lines 49-66);
- said local service agent checks whether or not there is a program of an application agent in question in an agent repository path of an agent repository of a base in execution; if there is any such program of the application agent, said local service agent asks the agent generating unit to load the application agent program from the agent repository to execute the application agent (col. 15, lines 36-54);
- in case of non-coincidence between the current base stored in the current directory storage unit with the base in execution actually executing the shell agent, said local service agent uses said remote call module to ask the local service agent to execute the external agent base (col. 7, lines 61-67, col. 8, lines 1-12 and col. 12, lines 49-66); and
- wherein said local service agent checks whether or not there is any program of an application agent in question in an agent repository path of the agent repository of the current base; if there is any such program of the application agent, said local service agent asks the agent generating unit to load the program of the application agent from the agent repository to execute the application agent (col. 15, lines 36-54).

As per claim **31**, Putzolu discloses:

- wherein there is provided a general-purpose application agent as said application agent; an extension application accommodating table memorizing and holding the relation between an extension and the application corresponding to said extension is referenced and retrieved from the extension of a designated file to boot an application having an extension coincident with the extension of the designated file (col. 6, lines 63-67, col. 7, lines 30-60 and col. 9, lines 48-56).

Response to Arguments

3. Applicant's arguments with respect to claims 1-31 have been considered but are moot in view of the new ground(s) of rejection.

Conclusion

4. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

U.S. Pat. No. 6,163,794 to Lange et al

U.S. Pat. No. 6,115,736 to Devarakonda et al

Any inquiry concerning this communication or earlier communications from the examiner should be directed to LaShonda T. Jacobs whose telephone number is 703-305-7494. The examiner can normally be reached on 8:30 AM - 5:00 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Ario Etienne can be reached on 703-308-7562. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is 703-305-3900.

LaShonda T. Jacobs
Examiner
Art Unit 2157

ltj
April 30, 2004

ARIO ETIENNE
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100